

Doing Systems Engineering Without Thinking About It at NASA Dryden Flight Research Center

Marta Bohn-Meyer
NASA Dryden Flight Research Center
P.O. Box 273 M/S 2009
Edwards, CA 93523
661-276-3199
marta.bohn-meyer@nasa.gov

Stephen Kilp
Science Applications International Corporation
NASA Dryden Flight Research Center
P.O. Box 273 M/S 4830C
Edwards, CA 93523
661-276-5280
stephen.g.kilp@saic.com

Peggy Chun
NASA Langley Research Center
NASA Dryden Flight Research Center
P.O. Box 273 M/S 2701
Edwards, CA 93523
661-276-5335
peggy.a.chun@nasa.gov

Abstract—When asked [1] about his processes in designing a new airplane, Burt Rutan responded:

...there is always a performance requirement. So I start with the basic physics of an airplane that can get those requirements, and that pretty much sizes an airplane... Then I look at the functionality... And then I try a lot of different configurations to meet that, and then justify one at a time, throwing them out... Typically I'll have several different configurations... But I like to experiment, certainly. I like to see if there are other ways to provide the utility.

This kind of thinking—engineering as a total systems engineering approach—is what is being instilled in all engineers at the NASA Dryden Flight Research Center.

TABLE OF CONTENTS

1. INTRODUCTION	1
2. DRYDEN FLIGHT RESEARCH CENTER BACKGROUND ..	3
3. HISTORY	3
4. PERCEIVED PROBLEM	5
5. SYSTEMS ENGINEERING QUESTIONS	7
6. HOW TO INSTILL SYSTEMS ENGINEERING THOUGHT PROCESSES INTO EVERYDAY ENGINEERING	9
7. CONCLUSIONS	14
8. ACKNOWLEDGMENTS	15
9. ACRONYMS	15
REFERENCES	16
BIOGRAPHIES	16

1. INTRODUCTION

The following is a basic systems engineering (SE) definition modified from NASA SP-6105, “NASA Systems Engineering Handbook” [2] with ideas from J. N. Martin’s, “Systems Engineering Guidebook” [3].

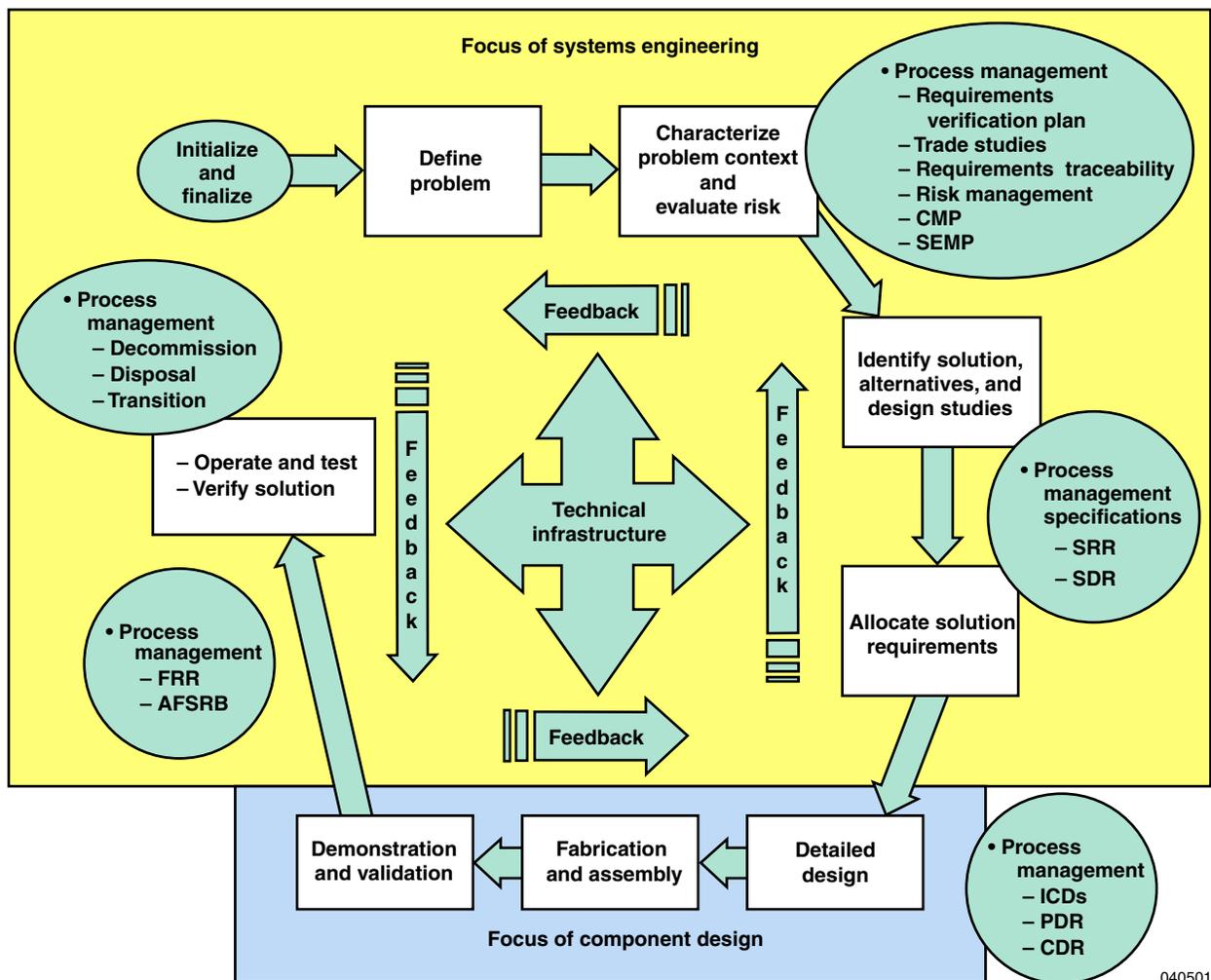
Systems engineering is an engineering approach controlling technical system development. It is an integrated approach to the design, development, evaluation, operation, and disposal of systems. The approach consists of identification and quantification of system requirements and goals, creation of alternative system design concepts, performance of trade studies, selection and implementation of the best design, verification that the design is properly built and integrated, and post-implementation assessment of how well the system meets requirements. The objective of a systems engineering approach is to ensure that an optimum balance of all system elements is achieved to meet the customer needs while optimizing the effectiveness, affordability, and safety of the system. The approach is usually applied repeatedly and recursively, with several increases in resolution of the system baselines (which contain requirements, design details, verification procedures and standards, as well as cost and performance estimates).

Although there is discussion within systems engineering communities on whether systems engineering is a way of thinking or a separate engineering discipline [4], the present evolution at Dryden Flight Research Center (DFRC)

is towards a way of thinking based on formal technical discipline, enhanced with on-the-job training (OJT) and additional coursework. At DFRC, the goal of systems engineering is an ability to think in terms of and to see the *big picture* with interactions and interfaces instead of only viewing individual and unique *parts* separately. This is not to say it is a vague or ambiguous activity. It is practiced by senior engineers who have extensive engineering backgrounds across several disciplines. Experience has come from OJT, as well as continuing updates to sharpen and maintain skills through formalized coursework across varied technical disciplines. Systems engineering is not a clear-cut path defined by a recipe-style process for cookie-cutter success. Systems thinking requires a person have technical skills and field experience; develop an organized framework for project implementation; have a systematic approach; be able to perform technical trade-offs and negotiations

throughout the life cycle of a project, task, or effort as it evolves from the initial definition phase; and have an understanding of the basic requirements and concepts to ultimately implement a solution achieving those requirements. Technical personnel at DFRC have not yet achieved all of these capabilities, but the implementation has been initiated. This paper describes the elements that are in place and the road that DFRC follows.

Systems engineering is the *enabler* supporting a cooperative team effort to achieve a pre-determined goal. Systems engineering supports the teaming and integration of all disciplines represented by engineers and technical support personnel on a project. At any particular level and definition of *system*, every engineer is a systems engineer. Figure 1 shows the basic systems engineering process at DFRC demonstrating this organization and their approach.



040501

Figure 1 - Basic Systems Engineering Process

2. DRYDEN FLIGHT RESEARCH CENTER BACKGROUND

Flight research separates “the real from the imagined,” and makes known the “overlooked and the unexpected.”
—Hugh L. Dryden

The NASA web site describes DFRC at www.dfrc.nasa.gov as follows:

Dryden Flight Research Center, located at Edwards, California, is the primary NASA installation for flight research. Projects at DFRC over the past 50 years have led to major advancements in the design and capabilities of many civilian and military aircraft.

The history of DFRC is the story of modern flight research in this country. Since the pioneering days after World War II, when a small, intensely dedicated band of pilots, engineers, and technicians dared to challenge the “sound barrier” in the X-1, DFRC has been on the leading edge in aeronautics and, more recently, in space technology. The newest, the fastest, the highest—all have made their debut in the vast, clear desert skies over DFRC.

It is within the context of this flight test environment and of this historically rich, flight experimental background that this paper discusses systems engineering. The factors that separate flight testing from all other types of tests are requirements to achieve robustness, and to assemble experimental techniques and ideas that achieve safe and successful results. The flight test environment offers an opportunity to build upon the results of empirical data, ground test data, and computer simulations exposing an experiment or a complete system to the ultimate reality of the flight environment. The dynamics of this flight environment cannot be created using only static and ground-based techniques and facilities. To achieve this flight environment, a wide range of techniques and capabilities are used getting as close as possible to the actual flight effects upon the experiment or system. The ability to effectively use these techniques distinguishes the uniqueness of the systems engineering capability at DFRC in conducting flight testing. Requirements for conducting a safe and successful flight test can include assessment and integration of all relevant information and test data, incorporation of all information and characteristics into a real-time simulation of the flight experiment, and determination and design of a suitable test bed (using various types of existing, modified, and novel flight vehicles) to probe the limits of physical boundaries. Systems engineering at DFRC ensures that the flight test is robust and is conducted safely and successfully. Robustness allows a capability to *fly, fix, and fly*, obtaining the most data

for each experiment or system. With robustness, new knowledge gained from the test can be incorporated and experiments or systems can be flown repeatedly. It is requirements such as these (imposed upon systems engineers at DFRC for all flight testing) that distinguishes systems engineering in a flight test environment.

At DFRC, the systems engineering role is not necessarily vested in one person or position. Depending on the size and complexity of the project, at DFRC, there may be a triad assigned to a project, which conducts the systems engineering role cooperatively. This triad is formed by the Project Manager, the Project Research Chief Engineer and the Project Flight Operations Engineer. The systems engineering role in this discussion encompasses just such a DFRC triad. This triad highlights the unique nature of the flight test environment and, as the primary NASA installation for flight research, the DFRC mission is to “conduct safe and timely flight research for discovery, technology, development, and technology transfer for U.S. Aeronautics and Space Preeminence.” The cooperative effort of challenging technical boundaries in the flight environment makes the expertise vested in the individuals of this triad crucial to the success of such projects.

3. HISTORY

Systems engineering is not new, it has been practiced as long as artifacts have been created and traded, and services provided [4, 5]. It has been defined and more progressively formalized in recent decades. It is still an evolving and maturing discipline.

- When did systems engineering begin?
- Who were the first systems engineers?
- What were the first systems engineering designs?

Scholars and scientists generally agree that systems engineering began as far back as 4000 BC. Rear Admiral Grace Hopper is quoted as saying [6], “Life was simple before World War II. After that, we had systems.” She was referring to the ever more complex systems that were being developed, even though the early “simple” style was also composed of unique items which formed systems. However, even the most basic of systems must integrate with the *world* to operate within its real-world constraints.

Systems Engineering has been around a long time. Tools, methods and methodologies have always been important and are continually evolving. The objective remains the same today as in the past: to achieve the big picture.

Notable systems engineering projects are listed in the following tables.

Table 1 - Ancient Systems Projects

Year	System
4000 BC	Water distribution systems in Mesopotamia
3300 BC	Irrigation systems in Egypt
400 BC	Urban systems, such as in Athens, Greece
300 BC	Roman highway systems

Table 2 - Modern Systems Engineering Projects

Year	System
1800s	Water transportation systems like the Erie Canal
1877	Telephone System (Considered by most SE historians to be the most significant SE accomplishment)
1800s	Electrical power distribution systems
1958–1972	Space systems programs
1958–1963	Mercury
1965–1966	Gemini
1963–1972	Apollo

Table 3 - Modern origins of Systems Engineering Approaches

Year	System
1944–1954	Western Electric and Bell Telephone Laboratories support to the Nike missile defense system development ¹
1951–1980	SAGE air defense system defined and managed by Massachusetts Institute of Technology ²
1954–1964	Atlas intercontinental ballistic missile program managed by systems contractor Ramo-Woolridge)
1968–1993	U.S. federal government-funded research to develop the Internet

¹NIKE [7] The initial Nike system, the Nike Ajax, was designed to supplement and then replace gun batteries deployed around major U.S. urban areas and military installations. Nike was named for the ‘Winged Victory’ Goddess of Greek mythology. The Nike missile batteries, or missile bases, consisted of three principle areas: the administrative area, integrated fire control area, and the launch area.

²SAGE [8] the Semi-Automated Ground Environment, was an automated control system for collecting, tracking, and intercepting enemy bomber aircraft used by NORAD from the late 1950s into the 1980s. It is generally considered to be one of the most advanced and successful large computer systems ever developed, especially for its day. By the time it was fully operational the Soviet bomber threat had been replaced by the Soviet missile threat, for which SAGE was entirely inadequate. Nevertheless, SAGE was tremendously important; it led to huge advances in online systems and interactive computing, real-time computing, and data communications using modems.

4. PERCEIVED PROBLEM

Systems Engineering conducted by a DFRC triad requires a great deal of coordination across disciplines and seems to face almost infinite possibilities for design trade-offs, schedule changes, requirements creep, and studies across the components.

Unique challenges at DFRC that promoted doing systems engineering without thinking about it result from the broad scope of different types of projects implemented there. These projects are characterized by a diverse range of DFRC roles and responsibilities on projects. The projects that generate these diverse roles for DFRC can include the following:

- (1) Internal DFRC-generated projects in which DFRC has full scope management of the project from initiation through implementation.
- (2) Externally initiated projects, in which DFRC can be a full partner in the early decision making and requirements and implementation tasks.
- (3) Externally initiated projects, in which DFRC only implements the final testing without input to initial planning or decisions.
- (4) Projects in which DFRC is a host only, and are completely managed by another organization. In a host-mode, DFRC may only provide hangar support, but still must understand the effort needed to assure safety on the premises for the project.

For example, the challenge of obtaining a flight test article, one which DFRC may or may not have input to requirements, which may or may not have addressed flight test issues and flight safety, requires a monumental effort on the part of an entire DFRC project team to implement successfully. The systems engineering triad needs to understand what the strengths and abilities are of each other, enhancing their interface internally so there is no communication or technical gaps in the systems engineering approach. The triad must operate as one entity, combining their unique expertise rather than operating as individuals.

Some commonly reported general problems of systems engineering (experienced by all organizations at one time or another) are as follows

- A lack of awareness of the importance, value, timing, accountability, and organizational structure of SE on programs
- The general unavailability of adequate, qualified resources within government and industry for allocation on major programs
- Insufficient SE tools and environments to effectively execute SE on programs
- Inconsistent or ineffective application of requirements definition, development, and management
- Poor initial program formulation
- A lack of the coordination across disciplines required for effective systems engineering.
- The large number of and wide range of possible design trade-offs across components
- A mutual distrust and lack of understanding that can occur across or between engineering disciplines
- The demand that systems be designed to last many years in a rapidly changing environment

The complexity of any technical project can be illustrated using an iceberg analogy (Figure 2). As with any technical project at DFRC the main focus is always on the test execution (the tip of the iceberg) part of the project. There is an overwhelming tendency to forget the multitude of tasks that it takes to support the test (illustrated by that part of the iceberg under the water).

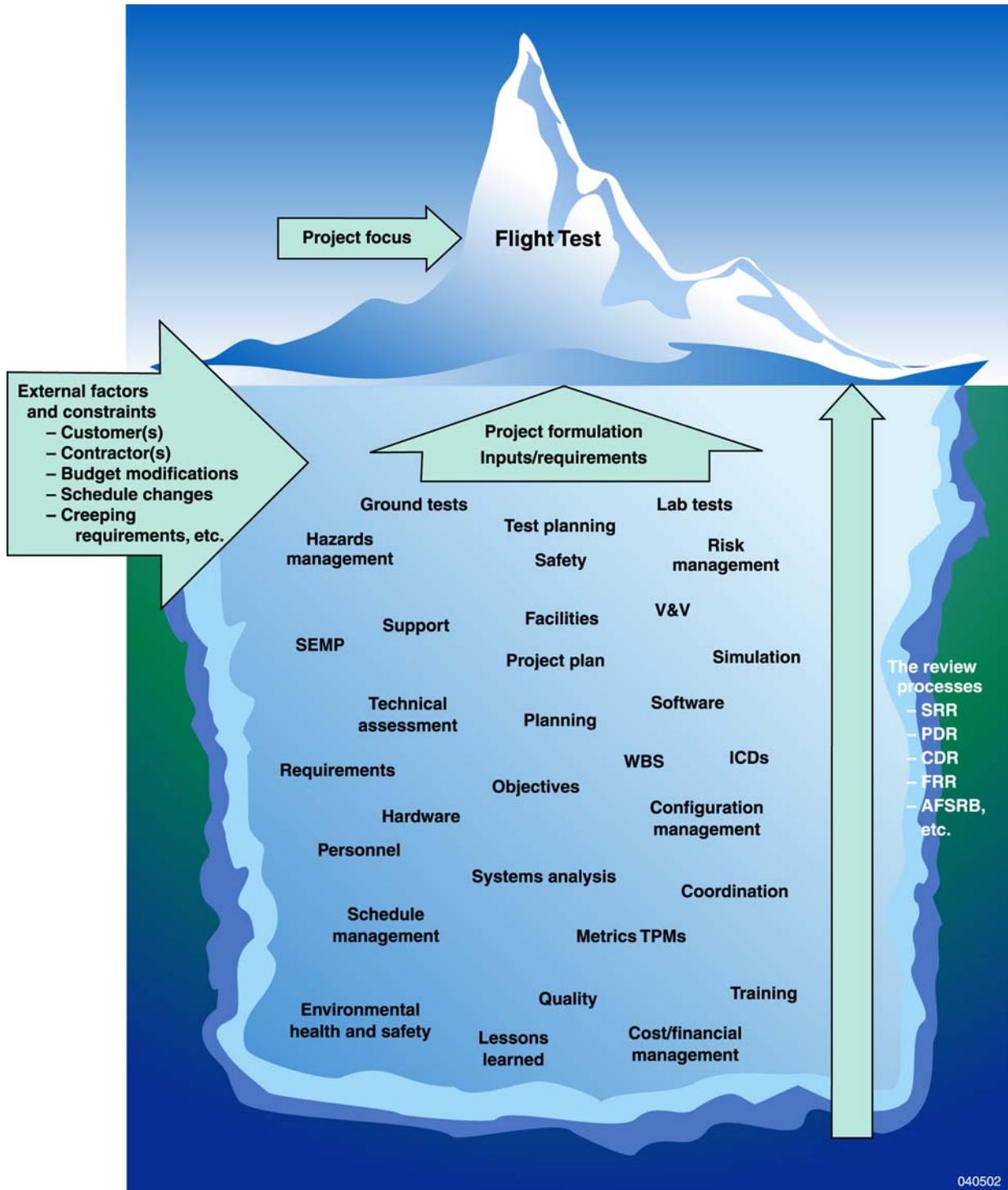


Figure 2 - The DFRC Project Iceberg

Taking a systems approach to project formulation starts at the top-level requirements (tip of the iceberg-flight test) and ensures that the final delivered project requirements meet these expectations. It is the part of the iceberg under water that all the engineers must focus on to use systems engineering effectively. Their systems thinking must guide the process as well as make use of their technical know how for the specifics of the design, analysis, and evaluation of solutions and alternatives.

Not paying attention to the *big picture* has direct implications on the success of the project. However, there are broader societal implications that history has provided to bring emphasis to the importance of the *big picture*. L. R. Graham [9] described the fate of an early twentieth century Russian systems engineer, Peter Palchinsky. Palchinsky campaigned for engineers to be responsible for the big picture, opposing the traditional role of Russian engineers, which was one of only solving specific technical problems brought to them by higher authorities.

Graham documented Palchinsky's writings from the mid 1920s. Palchinsky wrote that engineers should provide economic and industrial planning as well as technical expertise. The Soviet Union developed massive plans for technological modernization. There were Five-Year Plans developed for some of the most ambitious and gigantic technological projects, which did not incorporate Palchinsky's philosophy. For example, Palchinsky thought that engineers asked to design a large hydroelectric dam on a certain river should address a broad spectrum of issues that had far-reaching consequences when decisions were being made for projects of this size.

- What is the purpose of the dam and plant?
- Is this the best solution?
- What are the trade-offs among the alternatives, such as building a number of smaller plants versus one gigantic plant?
- Are resources available locally to run the plant?
- Will the energy be "transportable" to the users within a minimum distance from where it is generated?
- What impact will this have on the environment and the people who live in the area?

In order to answer these and other relevant questions an analysis and trade-off which includes technical, economic, social, and environmental effects of each has to be weighed. Peter Palchinsky was executed in 1929 for his views on engineering. Afterwards, the education of Russian engineers became very narrow.

In the book Graham related a 1960s experience he had:

I met a young woman who said that she was an engineer. 'What kind of an engineer?' I asked. 'A ball-bearing engineer for paper mills,' was the reply.

I responded, 'Oh, you must be a mechanical engineer.'

She rejoined, 'No, I am a ball-bearing engineer for paper mills.'

Incredulously I countered, 'Surely you do not have a degree in 'ball-bearings for paper mills.' She assured me that she indeed did have such a degree.

The rulers of the former Soviet Union also had narrow educational backgrounds. Between 1956 and 1986, the percentage of Politburo members with degrees in technical areas rose from 59 to 89 percent. Graham suggests that this narrowness of education had a lot to do with the disintegration of the former Soviet Union.

This certainly gives deeper and consequential meaning to Terry Bahill's question [10] to practicing systems engineers: "If you were arrested for being a systems engineer, could they gather enough evidence to convict you?"

5. SYSTEMS ENGINEERING QUESTIONS

The following provide some introspective questions to enhance the discussion points and fuel thoughts on systems engineering.

What is Systems Engineering?

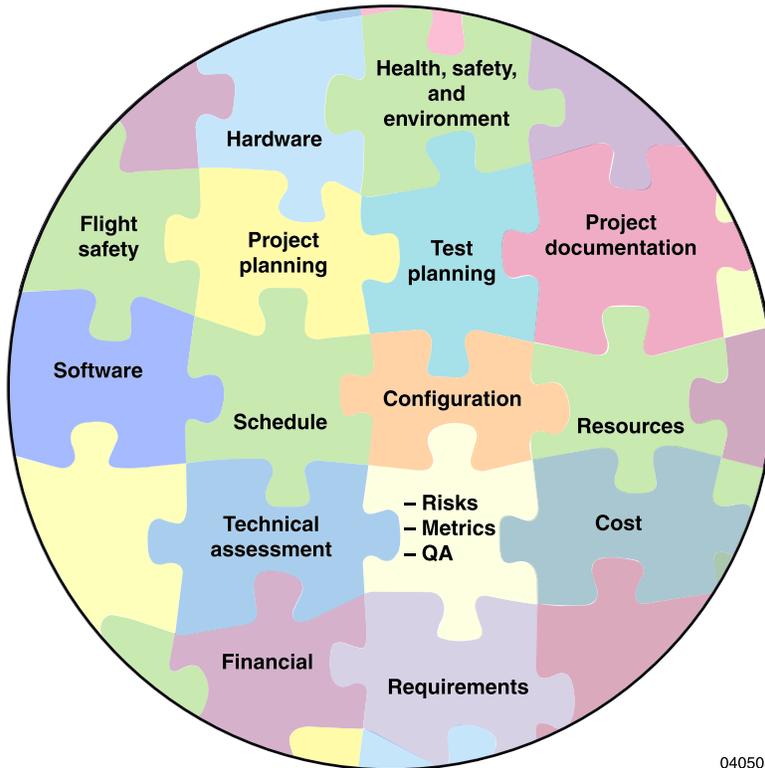
The definition of systems engineering used for this paper and modified from references 2 and 3 is as an engineering approach and process to control technical system development. It is an integrated approach to the design, development, evaluation, operation, and disposal of systems. A typical DFRC project can be thought of as separate jigsaw pieces of this complex process (Figure 3) that are composed of very diverse disciplines (represented by each different piece). These pieces of engineering discipline are assembled by the systems engineer to form the "system's puzzle" as depicted in figure 3.

What is the DFRC Systems Engineer Triad?

The systems engineer triad (composed of the Project Manager, the Project Research Chief Engineer, and the Project Flight Operations Engineer), shown in Figure 4, assembles the puzzle. In order to achieve a perfectly assembled puzzle, the triad ensures that all these pieces fit without friction and with enough tolerance taken into consideration.

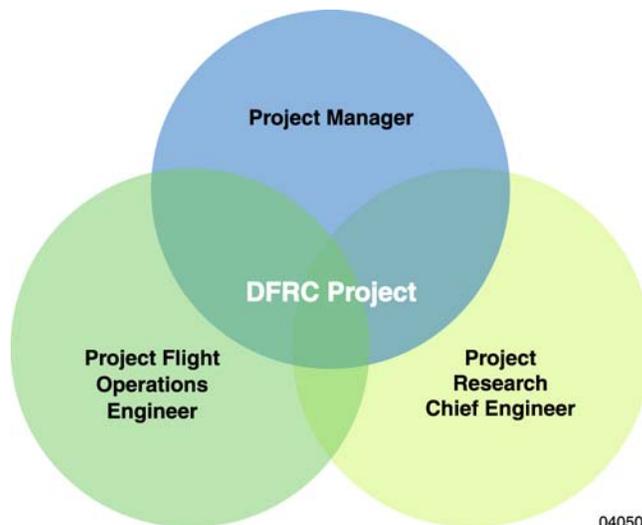
What Does the DFRC Systems Engineer Triad do?

The systems engineer triad becomes the complete-engineer and must do a little bit of everything. The systems engineer triad has to ask the right questions and determine that the answers are the right ones. The triad has to work together seamlessly to guide the technical effort on the project.



040503

Figure 3 - The DFRC Systems Engineering Puzzle



040504

Figure 4 - The DFRC Systems Engineering Triad

What Should the DFRC Systems Engineer Triad Know?

As the complete-engineer the systems engineer triad needs to know a little bit of everything, as well as capitalize on individual expertise in resource management, research system design, research integration and operations, and test bed integration, as well as being able to assess that each discipline engineer understands their own tasks and capabilities.

What is Important About the Role of the DFRC Systems Engineer Triad?

- Being able to communicate with each other, throughout all levels of the project and to all customers and stakeholders of the project
- Representing DFRC in the diverse project roles that DFRC implements
- Understanding real requirements (objectives, goals, requirements, processes, and specifications) in the context of the problem to be solved and the implications in the project life cycle
- Dealing with and managing changes internally and externally
- Being as knowledgeable as possible, but also learning to use available experts internal and external to DFRC

6. HOW TO INSTILL SYSTEMS ENGINEERING THOUGHT PROCESSES INTO EVERYDAY ENGINEERING

The DFRC systems engineer triad must focus on the tools, processes and methods that can promote a complete solution of the problems, not only specific solutions of specific problems. Other engineering disciplines concentrate on using their knowledge of the real world engineering elements (e.g., electrical circuits, flight controls, materials, robotics) and focus on finding solutions to the particular problems in their field. Figure 5 illustrates the standard DFRC project flow of requirements and how the Project Manager, the Project Research Chief Engineer, and the Project Flight Operations Engineer work together integrating systems engineering tools and processes to accomplish the goals of the project. Here again, there is not a solid boundary between the specific tasks attributed to the Project Manager, the Project Research Chief Engineer, and the Project Flight Operations Engineer. The boundary between the responsibilities is often fuzzy and should allow for interchange, negotiation, discussion, and resolution of issues, depending on the personality and character of the project requirements.

The DFRC conducts flight investigations of new aerodynamic configurations, high performance and highly maneuverable aircraft concepts, flight-crucial flight control systems, aircraft automation concepts, advanced propulsion systems and propulsion controls, advanced aircraft structural concepts, and flying qualities of highly augmented aircraft. The Research Engineering Directorate develops state-of-the-art flight measurement systems and flight test techniques needed to safely achieve the DFRC mission. Figure 6 illustrates how the research engineering directorate interfaces through an integrated product team (IPT) for a typical DFRC project.

Examples of on-going research projects are as follows:

- Intelligent Flight Control Systems
- Active Aeroelastic Wing
- Solar-Power Research (Pathfinder, Helios, etc.)
- ERAST (Environmental Research and Sensor Aircraft)
- Sonic Boom Research
- X-37 ALTV (Approach and Landing Test Vehicle)
- X-43 Hypersonic Research Aircraft

Project Research Chief Engineers are multidiscipline, in that they are both experimental development engineers and test and evaluation engineers. The Project Research Chief Engineer is responsible for and qualified to:

- Know the state-of-the-art through acquiring, understanding, and using appropriate reference materials and documents
- Generate knowledge through advancing the state-of-the-art in their specialty areas
- Publish and disseminate research and development results to the technical community through peer-reviewed technical publications, participation in technical conferences, and informal personal interaction with other members of the technical community

The Project Flight Operations Engineer leads the effort to establish and manage aircraft configurations and the flight test beds based on requirements from the Project Research Chief Engineer.

The Project Manager is uniquely responsible for assuming that the DFRC Air Worthiness and Flight Safety Review Process is conducted throughout the life cycle of the project. It goes without saying that the rest of the triad supports the Project Manager in this effort.

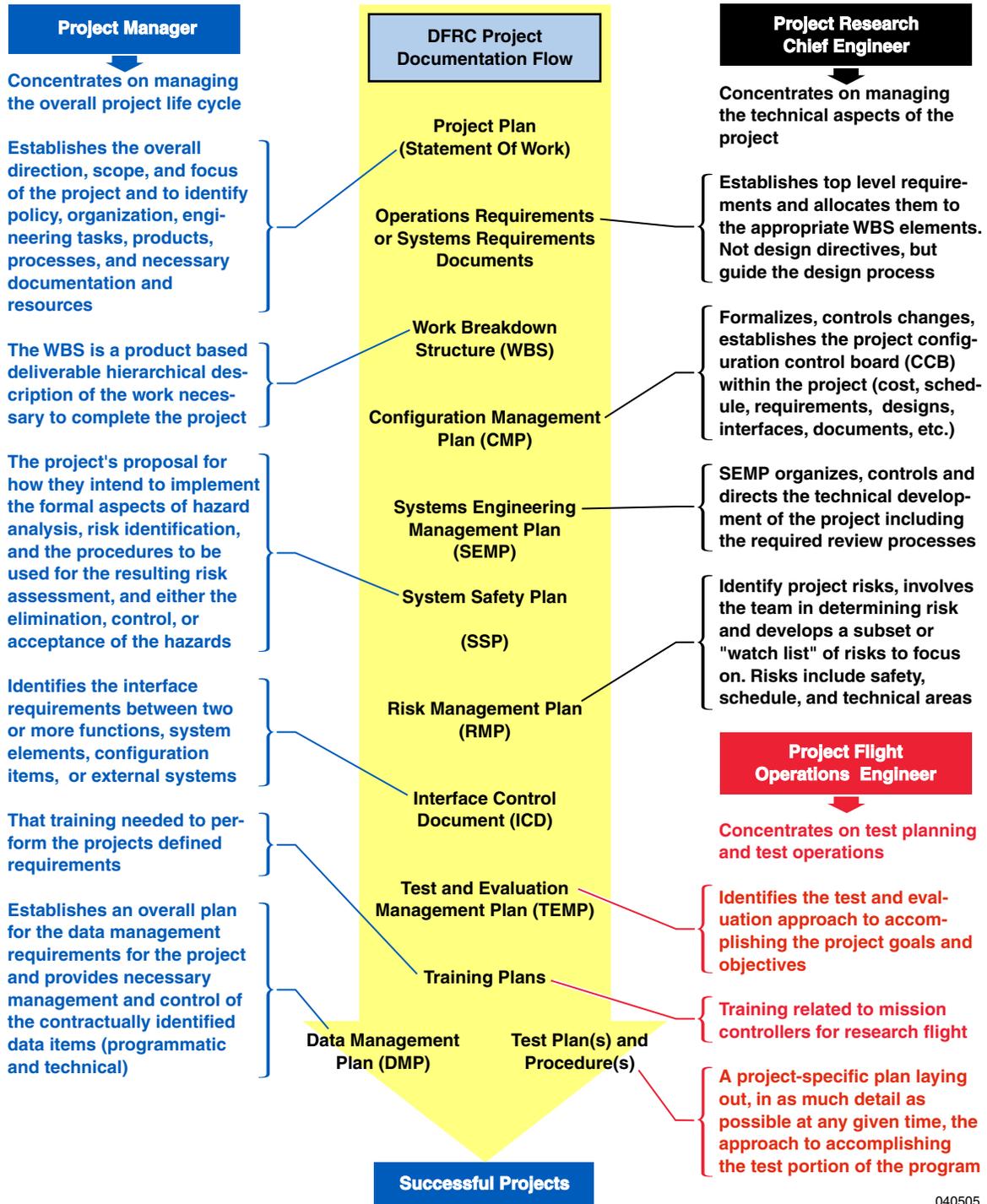
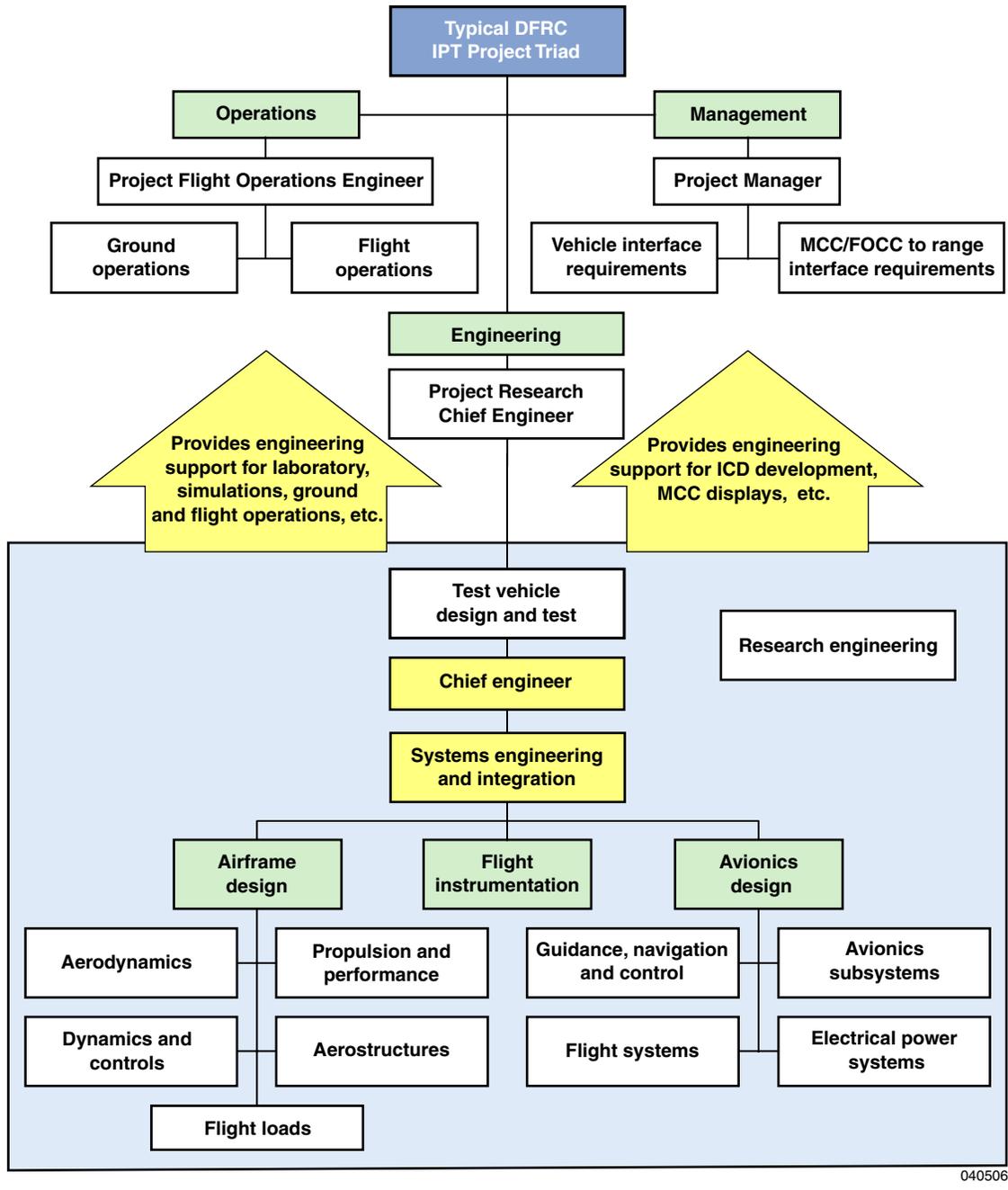


Figure 5 - Project Flow of Requirements through the Systems Engineering Triad at DFRC



040506

Figure 6 - Research Engineering and Typical DFRC Project IPT

Differences Between Large and Small Projects

A broad range of projects are supported at DFRC, from small, primarily in-house efforts to large, multicenter, multiagency efforts. The Systems Engineering triad approaches all of these efforts similarly, but assesses the specific systems engineering requirements based on the level of risk. For a relatively low-risk project, the fidelity, rigor, and formality of the systems engineering effort is modulated accordingly.

Training

You can train and educate; there needs to be on-the-job training.

There is a focus at DFRC to attract and retain a diverse, skilled, and professional workforce that possesses the competencies required to achieve the mission and goals of DFRC. The challenge is to train this workforce to handle the unexpected and unexplored, while at the same time maintaining the right mix of state-of-the-art competencies that can efficiently meet the NASA DFRC program requirements and still ensure challenging opportunities in a high-quality work environment.

The capability to provide quality training continuously is becoming a source of competitive advantage for organizations in both the government and the private sector. The most promising route for greater productivity lies in learning better and faster, thus improving each engineer's abilities to solve problems, innovate, and change.

Studies have indicated that traditional classroom training has produced few tangible productivity gains. These studies have found that an average of only 10 percent to 20 percent of formal training resulted in changing or enhancing ones performance on the job. Possibly a major reason is that classrooms artificially separate learning from real-world problems faced on the job. Adult learners are pragmatic—if the training isn't readily applicable to problems they deal with, they are likely to lose interest quickly. Another problem is that classroom training rarely is offered when it's needed in the fast-paced workplace of today.

This brings us to the question, how do we keep the systems engineer triad trained in state-of-the-art systems today? It is evident that formal training must be balanced with informal OJT and insight into best practices and cutting edge tools maintained. On-the-job training, formal training, simulations, and tools are critical for the DFRC systems engineering triad; so each person can understand their individual jobs, understand each others jobs and expertise, and understand the triad approach.

On the Job Training—“We learn from history that we do not learn from history.”

—Georg Wilhelm Friedrich Hegel, German philosopher

On-the-job training is one of the best training methods used at DFRC because it is conducted at the engineer's worksite by more experienced engineers (mentors). The most

common method used to broaden engineers' skills and increase their productivity is OJT. On-the-job training is important at DFRC when expertise is resident at DFRC, and formal training programs and resources are limited but an activity is a recurring technical task that a 'trainee' can expect to do many times throughout his/her career. To have a successful OJT program, supervisors assign a mentor to each engineer involved in OJT. The mentor has the responsibility to train carefully and monitor the development of the trainee.

The systems engineering triad approach is the key to understanding and passing on hard-earned lessons learned that are derived from the implementation of projects and programs. Perhaps the combination of formal training and OJT is the key that contributes to an ability to conduct systems engineering without thinking about it. An engineering discipline combined with black and blue marks to the psyche help to promote the organized, clear, and discerning thought process of an integrator for a project. Most project personnel have experienced first hand the frustration of repeating mistakes from previous programs, falling into the same pitfalls as predecessors, and failing to recognize and capitalize on hard-earned lessons learned. Few programs in today's environment of shrinking budgets and accelerated schedules can tolerate failure. In no place is this more true than in flight testing where mistakes carry both a heavy economic and political cost as well as the potential for loss of life. It is imperative that all projects and programs have access to the hard-earned lessons learned from flight testing through the years.

Many attempts have been made to create formal data bases that house these lessons learned. The Department of Defense was very successful in the incorporation of these lessons into well-documented military standards. These military standards did not implicate the projects guilty of mistakes, but, instead, applied the lessons learned in the form of general applications of best practices in specific technical areas. Unfortunately, these detailed military standards have been replaced with a broader application of commercial best practices. These new standards lose some of the mandated implementation requirements but provide more flexibility for tailoring, in the light of current project and program constraints. However, there are some voids in application and consistency.

The systems engineer triad provides a bridge in the interface across technical disciplines and subsystems on a project. The systems engineer triad also carries the burden of applying the lessons learned, understanding the best practices, and being able to recommend the required tailoring. The systems engineer triad has many resources available to sharpen the abilities to do systems engineering without thinking about it, but must carefully select the ones most applicable, leading the project from concept to reality, while making crucial decisions to challenge the boundaries of physics. These additional resources are 1) formal training, 2) simulations, and 3) tools.

Formal Training—When possible, OJT training must be supplemented by a formal training program. To sharpen systems engineering triad skills formal training and

experience means taking an interdisciplinary approach that enables the realization of successful systems. The focus at DFRC is on defining customer needs and required functionality, documenting requirements, synthesizing designs, implementing designs, verifying and validating systems, employing effective management techniques, and incorporating formal methods where that is feasible. Formal training includes introduction to NASA Project Management, NASA Systems Engineering, Carnegie Mellon University writing courses, Continuous Risk Management, Formal Methods, Earned-Value, Metrics, NASA Space Systems, and the Microsoft Project.

One way to acquire and document knowledge is through formal classroom instruction coupled with the learning potential that exists in the senior (experienced) engineer's everyday work. Much of this can be accomplished on a smaller, but probably more effective scale through familiar but underutilized techniques, such as cross training and sharing in teams, job rotations, developmental assignments, lessons-learned debriefings and action learning. New technology makes it possible to deliver "just-in-time" training to the desktop or, at the very least, to a central learning center located nearby.

The DFRC Research Engineering Directorate has embraced a training program in which senior engineers act as instructors for a very intensive one week Project Research Chief Engineer's course. In addition, a development Process and Training Program, is available to all at DFRC to develop individual career progression.

Incorporating systems engineering into this process either through on-the-job training or through formal training has really paid off in increasing the technical successes of projects conducted at DFRC.

Simulations—

Simulations are not the answer for understanding the real flight environment. Simulations are not a substitute for the actual flight environment; they are a substitute for no flight environment.

—Anonymous

Flight research is by nature unforgiving "of any carelessness, incapacity or neglect" (anonymous flight poster). Very often, it's impossible to predict what the unknown aviation and physical boundaries hold, even after painstaking examination of many formulas, models and simulations. That is why the result of a flight test is still the final answer when processing these predictions and best guesses down to two fundamental questions:

- Does it work?
- Does it work well enough to meet intended goals [11]?

Flight simulation and modeling are used extensively in support of the DFRC air vehicle development programs. The simulation tools are coupled with software-based design tools and the results of flight testing databases to mitigate development risks. Flight testing reduces the potential risk

for a project by "taking the flight data and making sense of it and updating the models so better fidelity of the model can be accomplished." It's a continuous feedback loop.

Often, as found at DFRC, testing a new system and its interaction with the existing systems of an airplane yield the most informative results. There is always the possibility that things won't work exactly as expected because of the complexity in modeling the old and new systems. Although many of the bugs can and should be worked out in the simulations, there is not yet a true substitute for flight testing.

At DFRC, two types of simulations for flight testing are used. The first is an analytical simulation, which consists of small segments of the overall flight with specific beginning and ending states. The segments are not readily linkable to represent and assess the overall flight. The segments allow valuable assessment of specific portions of the flight that may be critical to success. Monte Carlo techniques are typically used in this type of simulation. The second type of simulation is one of flight planning. This enables development and assessment of the overall flight control environment, since it is an actual model of the flight environment incorporating actual flight test data as it becomes available.

Simulation at best verifies for the engineer what they think they know about how the test vehicle should fly. However, simulation is a prediction based on (analysis and synthesis of) data, and is not always linear. It may not always fly as predicted or do what the simulation indicated. During actual flight test the pilot can tell the engineers what is really happening.

Simulation from a systems engineering perspective is useful for integrating all of the information, whether ground test, analogy, or analysis, etc. It provides the project team (triad) with an integrated *big picture* of how individual systems (perhaps developed independently) will *play together* during a flight test program.

The SE triad must appreciate the limitations of these simulation tools—being careful not to draw conclusions that are too rigid when multiple assumptions—not yet validated—are required to initialize the simulation. Tools in the spirit of the DFRC triad, are approached knowing constraints of limited time and budget, and are balanced with the usefulness to the project. Often, unless specifically required, simple or uniquely designed and tailored tools are used instead of larger tools which require significant support costs.

As Hugh Dryden said—we have to 'separate the real from the imagined.'

Tools

A fool with a tool is still a fool—Author unknown

A major part of achieving success through implementation of a systems engineering philosophy is the correct and

efficient use of resources, including tools, to enable and facilitate communication and the sharing of information across all facets of a project. There are many software tools designed for this. These tools range from complex and wonderful to simplistic and basic. Tools need to fit into the way the project operates versus driving the project to fit in with the tools. Therein lies a key issue which plagues many projects. Too often, a tool is selected because it is seen to be the ultimate solution to resolving complex problems. For example, just because you have a particular tool does not mean you will get the requirements correct without intensive effort and participation on the part of all of the team members, customers, and stakeholders through examining all possibilities for meeting the original goals and objectives of the project. A tool helps to organize, facilitates the communication, and serves as a repository or documentation of a project. A tool can facilitate the communication needed across organizations or subsystems; it cannot resolve the interface issues, assure that testing is conducted properly, or that assumptions are valid. The systems engineer triad together with the discipline engineers and all other project participants needs to provide the validation. The tool supports and organizes the information for the team.

In a paper by Masashi Mizukami [12], a tool was designed and demonstrated for small projects at DFRC. This tool was referred to as *electronic Systems Engineering* (eSE) and was piloted and documented at DFRC. This eSE software is based on an online systems engineering tool for flight research projects and was developed using a workgroup database. Capabilities are implemented for typical flight research systems engineering needs in: a document library, configuration control, hazard analysis, a hardware database, requirements management, action item tracking, project team information, and technical performance metrics. Repetitive tasks are automated to reduce workload and errors. Current data and documents are instantly available online, and can be worked on collaboratively. Existing forms and conventional processes are used, rather than inventing or changing processes to fit the tool. An integrated toolset offers advantages by automatically cross-referencing data, minimizing redundant data entry, and reducing the number of programs that must be learned. With a 'keep it simple' approach, significant improvements are gained over existing capabilities for minimal cost. By using a workgroup-level database platform, personnel closest to the project can develop, modify, and maintain the system, thereby saving time and money. As a pilot project, the system is being used to support an in-house flight experiment. An option is proposed for further developing and deploying this type of tool on a wider basis in the organization.

According to Mizukami:

Software tools often are used to facilitate systems engineering tasks, and these tools provide potential benefits. For example, current project data and documents can be instantly accessed online, and repetitive tasks can be automated, resulting in error reduction and improved situational awareness. A net savings of time and money could be realized, even considering the

upfront investment to implement the software tools.

In flight research, however, each project is technically and programmatically unique, so a standard set of software tools is often unavailable or not applicable. If enterprise-level software packages were implemented, the life cycle cost for procurement, development, training, and administration would be high and burdensome for a relatively small organization like the NASA Dryden Flight Research Center (Edwards, California). Furthermore, NASA Dryden frequently is a partner in a project led by another organization, in which the lead organization often mandates usage of its set of tools. NASA Dryden would then become a client user of those software packages, which is the proper and economical approach, but any sizable investments in in-house tools are not recouped.

This tool for in-house Dryden projects provides some of the basic ingredients for a useful project resource:

- (1) Interfaces with existing software systems.
- (2) Requires no expensive overhead support for the tool.
- (3) Enables on-line modifications of documentation and review without conflicting changes in real time.
- (4) Requires minimal training because uses current applications.

Some key features of the tool are the following:

- Allows team members to view and edit documents online
- Provides user ID and password access control
- Provides privilege control based on user access level and document status
- Allows electronic signatures
- Allows electronic attachments
- Generates summary data automatically

7. CONCLUSIONS

The DFRC systems engineering triad is on the path to performing systems engineering without thinking about it, with an unprecedented level of expertise, training, and abilities in understanding the subsystems that comprise this system. The triad is the combination of the Project Manager, the Project Research Chief Engineer and the Project Flight Operations Engineer. They each have to understand the systems engineering process and be able to interact as one total systems engineer. They have to understand their strengths and weaknesses to operate seamlessly for successful and safe technical integration of the project. There are a number of tools that they can use to develop as the

systems engineering triad and to hone their personal expertise. These include OJT and lessons learned, simulations and off-the-shelf and customized computer tools. The unique roles supported at DFRC on projects requires that the systems engineering triad have unique views into the systems engineering role and provide the expertise to perform flight test research to push the boundaries of physics using efficient and safe techniques and processes.

The approach to doing ‘systems engineering without thinking about it’ described in this paper is one created at DFRC. The systems engineering effort is a transparent one at DFRC. There is no identified ‘systems engineering’ activity. The DFRC systems engineering triad supports systems engineering as an activity that links across DFRC in a monitored continuum. This approach addresses the variety of projects at DFRC with the goal of being efficient and responsive in the application of systems engineering approaches on these projects and in using best practices and NASA level project implementation requirements as well as using the project work force effectively. The ultimate goal is to engrain and achieve this approach 1) across the board or 2) comprehensively at DFRC. Dryden is working on this and has levels and parts of it in place, including the triad, the basic requirements for technical discipline expertise, the emphasis on OJT, and continual technical training. DFRC is on this road to implementation of practicing systems engineering without thinking about it. Most of the elements are in place, with the last one in a draft form, an integrated systems engineering training, that crosses all engineering disciplines rather than treating SE as a distinct separate discipline. The ultimate integration of all of these elements is a final step in achieving systems engineering without thinking about it. This path at DFRC uses technical discipline, OJT, tools, and integrated systems engineering training to accomplish each of the following; integrate the triad, enable smooth communication, and assure that each member of the team understands the individual expertise and the power of the amalgamation, as well as understanding the individual expertise of the technical discipline engineers that the triad supports.

8. ACKNOWLEDGMENTS

The authors wish to acknowledge the Project Research Chief Engineers, Flight Operations Engineers and Project Managers at DFRC over the years who have, through their trials and tribulations, learned to appreciate the view of the “big picture” to successfully implement projects in achieving their requirements and the leadership of DFRC supporting the development and encouragement of those who strive to maintain the systems engineering way of thinking. Also, acknowledgment of one specific individual is required, who has provided guidance in systems engineering thinking at DFRC by example and contributions throughout his esteemed career and his continuing contributions to organizations including INCOSE and ITEA, Mr. Donald Greenlee of SAIC.

9. ACRONYMS

AFSRB	Airworthiness and Flight Safety Review Board
ALTV	Approach and Landing Test Vehicle
CDR	Critical Design Review
CMP	Configuration Management Plan
DFRC	Dryden Flight Research Center
DMP	Data Management Plan
ERAST	Environmental Research Sensor Aircraft
FOCC	Flight Operations Control Center
FRR	Flight Readiness Review
ICD	interface control document
INCOSE	International Council on Systems Engineering
IPT	integrated product team
ISO	International Standards Organization
ITEA	International Test and Evaluation Association
MCC	Mission Control Center
NASA	National Aeronautics and Space Administration
NORAD	North Atlantic Defense
OJT	on-the-job training
PDR	Preliminary Design Review
QA	quality assurance
RMP	Risk Management Plan
SAGE	Semi-Automated Ground Environment Air Defense System
SAIC	Science Applications International Corporation
SDR	System Definition Review
SE	systems engineering
SEMP	Systems Engineering Management Plan
SRR	System Requirements Review
SSP	System Safety Plan
TEMP	Test Evaluation Master Plan
TPM	technical performance measure
V&V	validation and verification
WBS	work breakdown structure

REFERENCES

- [1] "Talking with Burt," *Air and Space Magazine*, June 2004, Burt Rutan, Aircraft Designer, interviewed by Diane Tedeschi. Accessed December 4, 2004. Online at: <http://www.airandspace magazine.com/asm/index.html>
- [2] Dr. Robert R. Shishko, *NASA Systems Engineering Handbook*, NASA SP-6105, June 1995.
- [3] J. N. Martin, *Systems Engineering Guidebook: A Process for Developing Systems and Products*, CRC Press, Boca Raton, FL, November 1996.
- [4] Karl Arunski, James Martin, Phil Brown, Dennis Buede, "Systems Engineering Discipline" *INCOSE, WMA Chapter, History*. Presentation to the Texas Board of Professional Engineers, Sept 10, 1999. Accessed December 4, 2004. <http://www.incose-wma.org/info/se/SystemsEngineeringHistory/sld001.htm>
- [5] "Engineering History," *Project Lead The Way, Inc.*, Power Point Briefing, Accessed December 4, 2004. http://www.pltw.org/OCHM/POE/Lesson%201_1--History%20of%20Engineering.ppt
- [6] Philip Schieber, ed, "The Wit and Wisdom of Grace Hopper", *OCLC Newsletter*, No. 167, March/April 1987, Following a speech by Hopper at Ohio State University February 5, 1987. Accessed December 4, 2004. Online at: <http://www.cs.yale.edu/homes/tap/Files/hopper-wit.html>
- [7] The Nike Site, "Nike Ajax," from *Redstone Arsenal Historical Information Web site*. Accessed December 10, 2004. Online at: <http://www.redstone.army.mil/history/nikesite/nikeajax.html>
- [8] "SAGE," *Wikipedia, The free encyclopedia*, Accessed December 4, 2004. Online at <http://en.wikipedia.org/wiki/SAGE>
- [9] Loren R. Graham, *The Ghost of the Executed Engineer* First Harvard University Press, Cambridge, Massachusetts, 1993.
- [10] Terry Bahill and Gissing, "Re-evaluating Systems Engineering Concepts Using Systems Thinking," *IEEE Trans. on Systems, Man, and Cybernetics*, Part C, Vol. 28, No. 4, 1998, pp. 516 to 527.
- [11] Jay Levine, "Flight Research—the ultimate proof of concept." *The X-PRESS*, Special Edition, December 17, 2003, DFRC. Accessed December 4, 2004. Online at: http://www.dfrc.nasa.gov/Newsroom/X-Press/special_editions/AAW/index.html
- [12] Masashi Mizukami, *Computer-Aided Systems Engineering For Flight Research Projects Using a Workgroup Database*, NASA/TP-2004-212860, 2004.

BIOGRAPHIES

Marta Bohn-Meyer is currently the Chief Engineer for the NASA Dryden Flight Research Center (DFRC) and is responsible for reviewing all flight research projects conducted at NASA DFRC and recommending and approving readiness to commence flight activity. A member of the Federal Senior Executive Service Corps, she began her civil service career with NASA as an Operations and Flight Test Engineer and has held a number of engineering and management positions including Project Manager, Director for Safety and Mission Assurance, and Director for Flight Operations before assuming this post in September 2001. She has a BS in Aeronautical Engineering from Rensselaer Polytechnic Institute and an Executive MBA from Simmons College.

Stephen Kilp is a Development Test Engineer with the Science Applications International Corporation (SAIC) supporting DFRC in the systems engineering and integration office. He previously served as a senior level flight test engineer at the Air Force Flight Test Center, Edwards AFB, with extensive experience in technical and test management of complex aircraft weapon systems (B-52, F-5, F-16 and B2A). He has additional expertise in integrating environmental and safety related requirements into test operations, preparing key sections of technical proposals. He has a BS in Physics and an MS in Technical Management.

Peggy Chun is currently a systems engineer for the NASA Engineering and Safety Center (NESC) and is responsible for providing technical integration guidance and support for NESC activities. Her previous NASA activities included serving as the acting head of the Systems Engineering and Integration Office at Dryden Flight Research Center, responsible for supporting and infusing systems engineering and best engineering practices into the engineering organization. Prior to her NASA service, she worked at the Naval Air Warfare Center, China Lake, CA, in a variety of capacities including project manager, systems engineer, test engineer, and propulsion engineer, on diverse flight projects. She has a BA and BS in Physics from the University of Minnesota, Duluth, and an MS in Systems Engineering from California State University, Northridge.